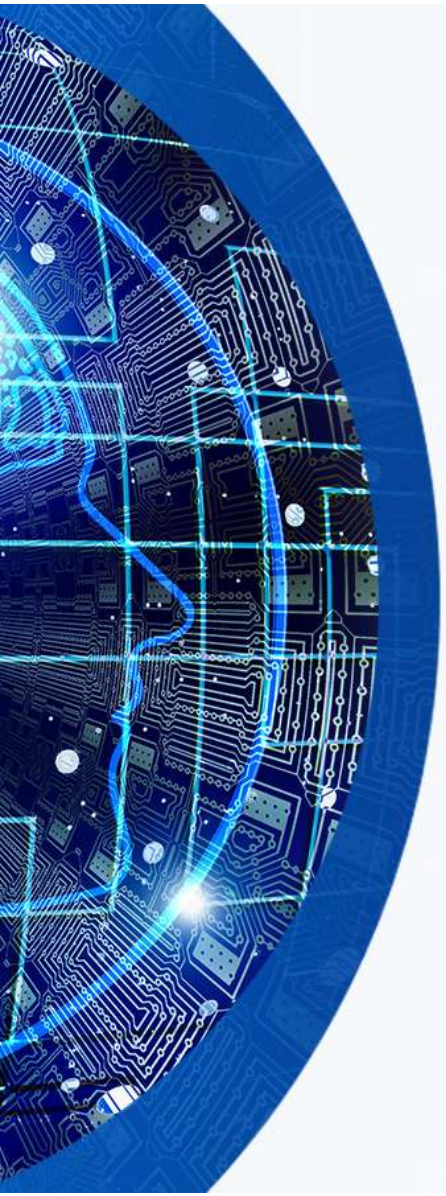
Deep Learning

By,
Lubna Rahman

What is Deep Learning?

- Subfield of Machine Learning that teaches computers to do what comes naturally to humans.
- Based on algorithm inspired by the structure and function of the brain called Artificial Neural Networks(ANN).
- It is essentially a neural network with three or more layers.
- Also called as large neural networks.



Applications of Deep Learning

- Automated Driving
- Medical Research
- Aerospace and Defense
- Electronics
- Industrial Automation
- E-commerce and Entertainment
- Facial recognition and Image colorization

Why Deep Learning ?

1. Large volume of labeled data

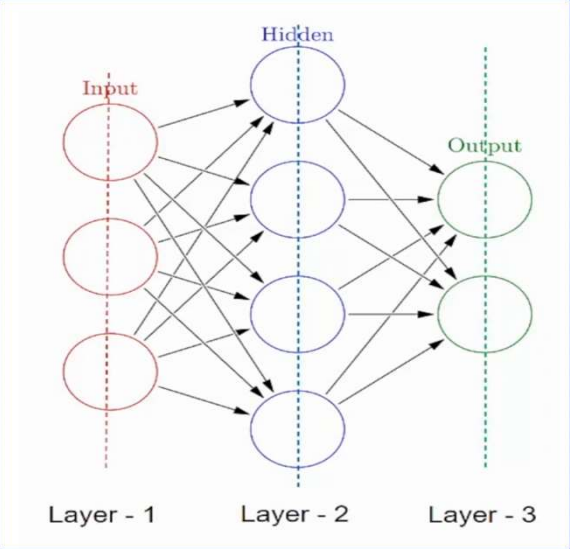
- Internet
- Social Media
- Business databases

2. Substantial computing power

- High performance GPU
- Cloud computing
- High performing tools

How Deep Learning Works?

- Deep learning methods use neural network architectures, hence are often referred to as deep neural networks.
- A neural network is made up of vertically stacked component called Layers.



Layers of Neural Network

1. *Input Layer*:
   - It accepts the data and pass it to the rest of the network.

2. *Hidden Layer*:
   - They are either one or more in number and are responsible for the excellent performance and complexity of neural networks.
   - They perform multiple functions at the same time such as data transformation, automatic feature creation and pass the result to the output layer.

3. *Output Layer*:
   - It recieves the result of the problem. This whole process is called Forward propagation.
   - Raw images are passed to the input layer and result is received in the output layer.

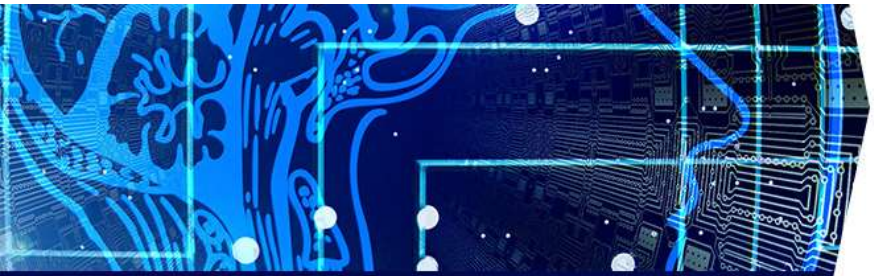
How Deep Learning Works?

In an Artificial Neural Network(ANN), signals travel between nodes and assign corresponding weights.

A heavier weighted node will exert more effect on the next layer of nodes. The final layer compiles the weighted inputs to produce an output.

Deep learning systems require powerful hardware because they have a large amount of data being processed and involves several complex mathematical calculations.

When processing the data, ANN are able to classify data with the answers received from a series of binary true or false questions involving highly complex mathematical calculations.



Activation Functions

Activation functions are functions used in a neural network to compute the weighted sum of inputs and biases, which is in turn used to decide whether a neuron can be activated or not.

They manipulate the presented data and produces an output for the neural network that contains the parameters in the data.

They are also referred to as transfer functions.

The Activation Functions can be basically divided into 2 types:

- Linear Activation Function
- Non-linear Activation Functions

Linear Activation Function

A linear function is also known as a straight-line function where the activation is proportional to the input i.e. the weighted sum from neurons. It has a simple function with the equation: $f(x) = ax + c$

The problem with this activation is that it cannot be defined in a specific range.

It makes the activation function work like linear regression. The final layer of the Neural Network will be working as a linear function of the first layer.

Another issue is the gradient descent when differentiation is done, it has a constant output which is not good because during backpropagation the rate of change of error is constant that can ruin the output and the logic of backpropagation.

Non-Linear Activation Functions

The non-linear functions are known to be the most used activation functions.

It makes it easy for a neural network model to adapt with a variety of data and to differentiate between the outcomes.

These functions are mainly divided basis on their range or curves:

- Sigmoid Activation Functions
- Tanh Activation Functions
- ReLU(Rectified Linear Units)Activation Functions

Non-Linear Activation Functions

Types

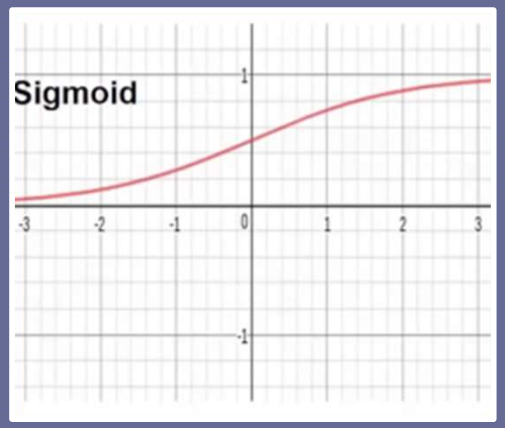
Sigmoid Activation Function: Sigmoid takes a real value as the input and outputs another value between 0 and 1. It translates the input ranged in $(-\infty,\infty)$ to the range in (0,1).

Tanh Activation Function: The Tanh will map values between -1 and 1. Tanh looks very similar to the shape of the sigmoid but it restricts the range between -1 and 1.

ReLU Activation Function: Despite its name, Rectified Linear Units, it's not linear and provides the same benefits as Sigmoid but with better performance.

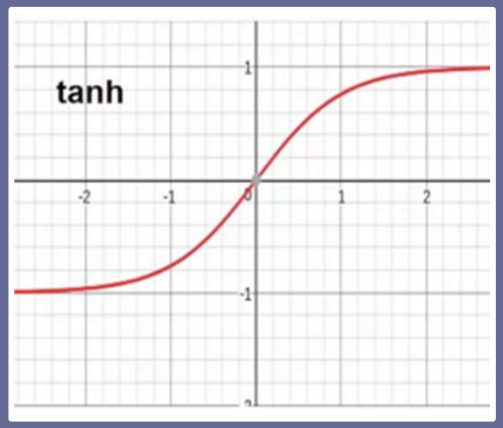
Non-Linear Activation Functions

Types



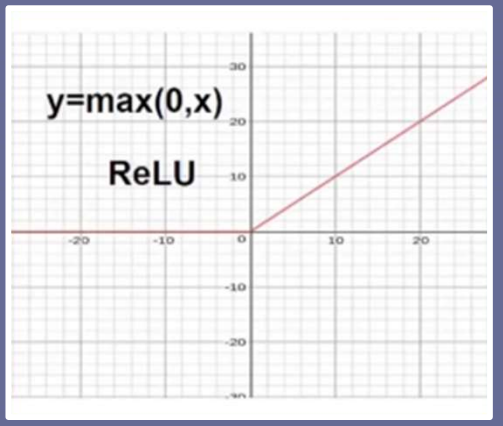
1

Sigmoid Activation Function



2

Tanh Activation Function



3

ReLU Activation Function

Which activation function to use?

Depends on the problem type and the value range of the expected output. For instance:

- To predict values that are larger than 1, ReLU is used as tanh or sigmoid are not suitable to be used in the output layer.
- If the output values have to be in the range (0,1) or (-1, 1) then ReLU is not a good choice so sigmoid or tanh are used.

Deep Learning at work

1. Facial recognition

- works by learning to detect , recognize edges, lines of faces and then more significant parts of the faces.
- Finally, detects the overall representations of faces and over time, the program trains itself thereby increasing the probability of correct answers.
- Facial recognition accurately identify faces with time.

2. Virtual Assistants

1. The virtual assistants like Alexa or Siri use deep learning to help understand the speech and language humans use when they interact with them.

Deep Learning at work

3. Vision for driverless delivery trucks, drones and autonomous cars

- Through deep learning algorithms, an autonomous vehicle understands the realities of the road and how to respond to them whether it's a stop sign, a ball in the street or another vehicle.
- The more data the algorithms receive, the better they are able to act human-like in their information processing—knowing a stop sign covered with snow is still a stop sign.

4. Chatbots and service bots

- respond in an intelligent and helpful way to an increasing amount of auditory and text questions with the help of deep learning.

Deep Learning Algorithms

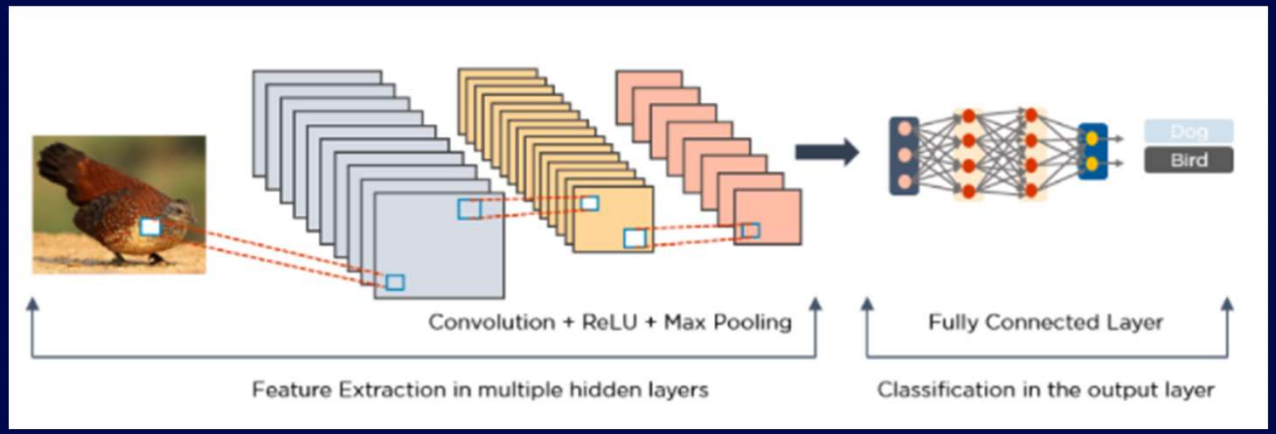
- Convolutional Neural Network (CNN)
- Recurrent Neural Network (RNN)
- Long Short Term Memory Network(LSTM)
- Back Propagation
- Stochastic Gradient Descent (an optimization algorithm)

CNN

- CNNs also known as ConvNets, consist of multiple layers and are mainly used for image processing and object detection.
- CNN has a convolution layer that has several filters to perform the convolution operation.
- CNN's have a ReLU layer to perform operations on elements. The output is a rectified feature map.
- Pooling: It is a down-sampling operation that reduces the dimensions of the feature map. The rectified feature map next feeds into a pooling layer.

CNN

- The pooling layer then converts the resulting two-dimensional arrays from the pooled feature map into a single, long, continuous, linear vector by flattening it.
- A fully connected layer forms when the flattened matrix from the pooling layer is fed as an input, which classifies and identifies the images.



RNN

- Recurrent Neural Network (RNN) is a type of artificial neural network which uses sequential data or time series data.
- They are commonly used for ordinal or temporal problems, such as language translation, natural language processing (nlp), speech recognition, and image captioning.
- incorporated into popular applications such as Siri, voice search, and Google Translate.
- Like convolutional neural networks (CNNs), recurrent neural networks utilize training data to learn

RNN

How Do RNNs work?

The output at time t-1 feeds into the input at time t.

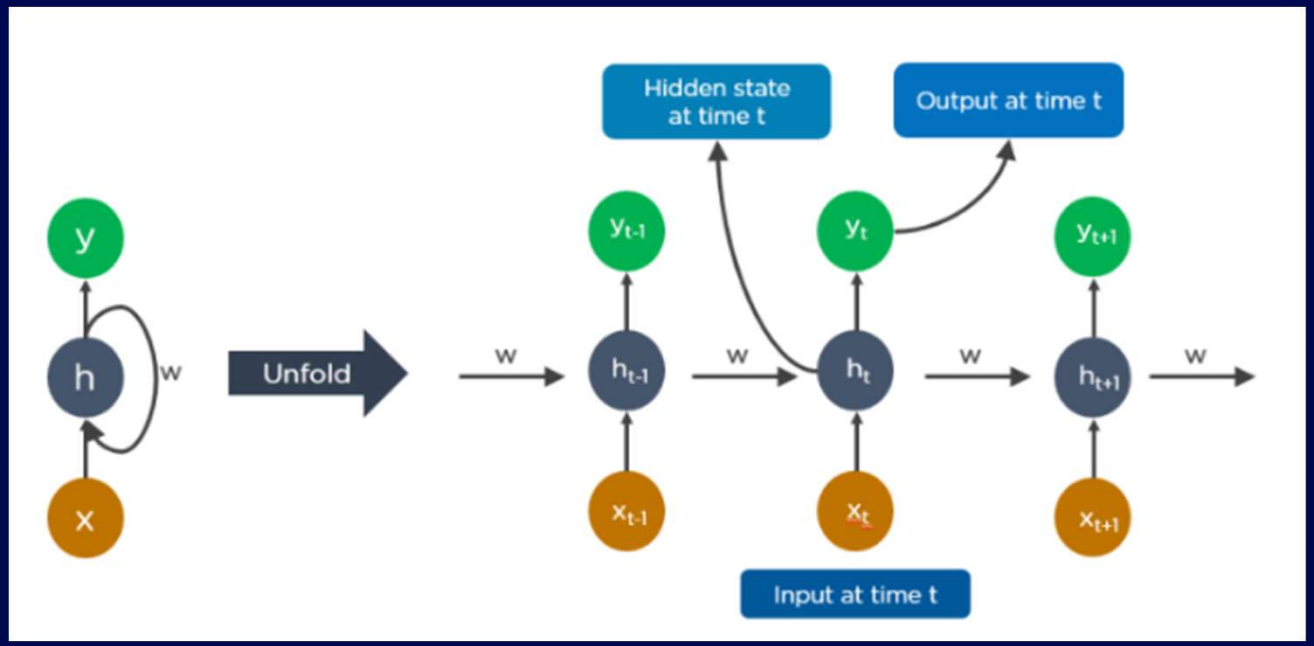
The output at time t feeds into the input at time t+1.

RNNs can process inputs of any length.

The computation accounts for historical information, and the model size does not increase with the input size.

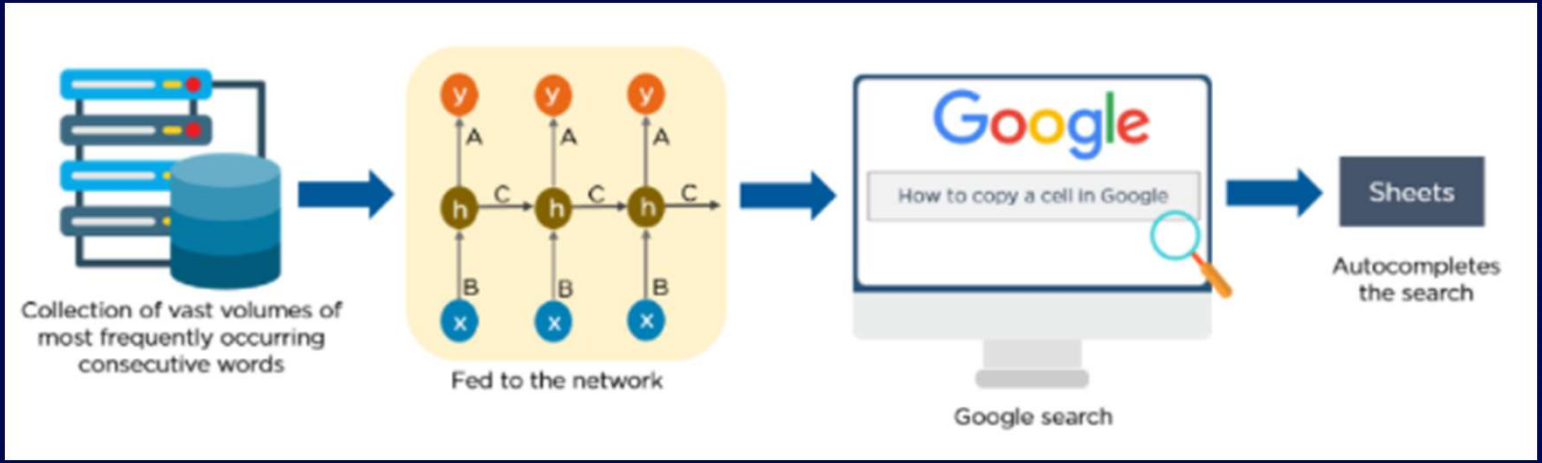
RNN

- An unfolded RNN



RNN

- An example of how Google's autocompleting feature works on RNN



LSTMs

LSTMs are a type of Recurrent Neural Network (RNN) that can learn and memorize long-term dependencies.

Retaining past information for long periods is the default behavior.

Useful in time-series prediction because they remember previous inputs.

LSTMs have a chain-like structure where three interacting layers communicate in a unique way.

Besides time-series predictions, LSTMs are typically used for speech recognition, music composition, and pharmaceutical development.

LSTM Architecture

How Do LSTMs Work?

The LSTM consists of three parts which perform an individual function:

The first part chooses whether the information coming from the previous timestamp is to be remembered or is irrelevant and can be forgotten

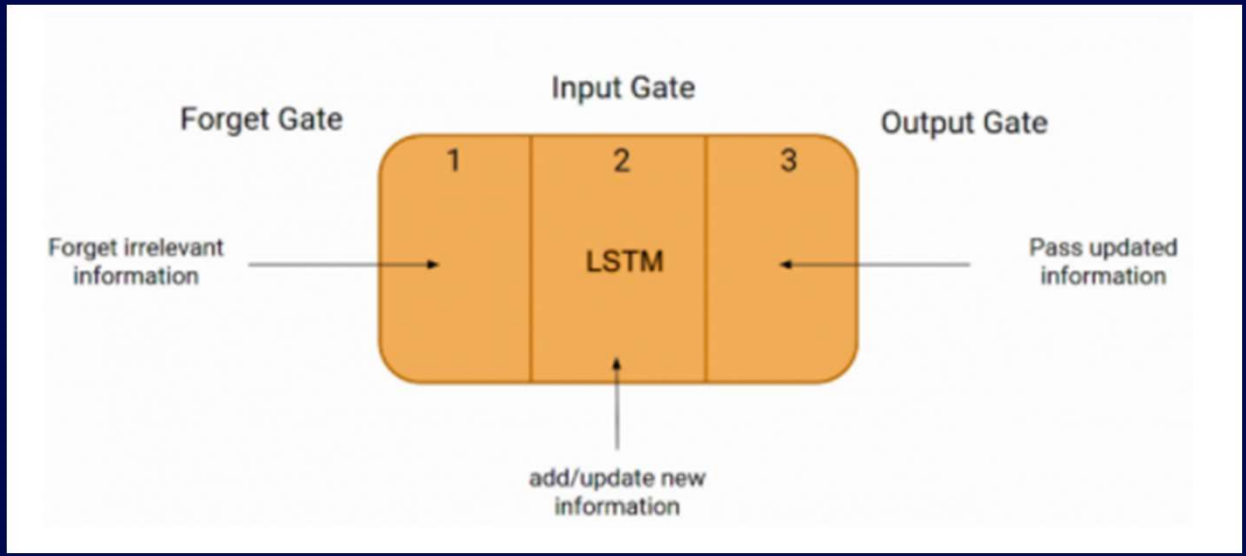
In the second part, the cell tries to learn new information from the input to this cell and they selectively update the cell values.

Finally, in the third part, the cell passes the updated information from the current timestamp to the next timestamp.

LSTM Architecture

Gates

- These three parts of an LSTM cell are known as gates. The first part is called Forget gate, the second part is known as the Input gate and the last one is the Output gate.



Backpropagation

Backpropagation algorithm is one of the most fundamental building block in a neural network.

Used to train a neural network through a method called chain rule.

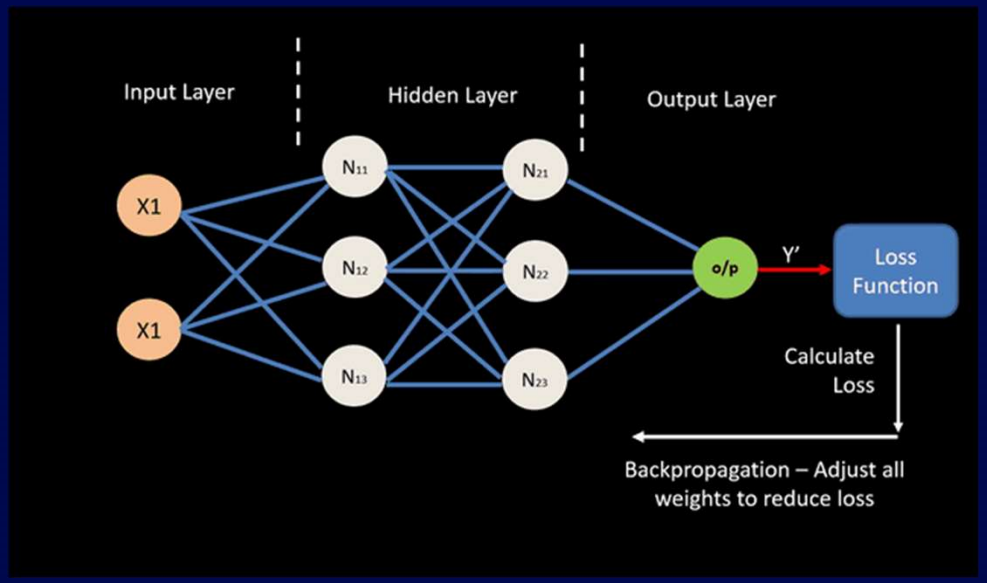
After each forward pass through a network, backpropagation performs a backward pass while adjusting the model's parameters (weights and biases).

Repeatedly adjusts the weights of the connections in the network so as to minimize a measure of the difference between the actual output vector of the net and the desired output vector.

Backpropagation aims to minimize the cost function by adjusting network's weights and biases

Backpropagation

- Backpropagation adjusts all the weights to minimize the loss function, calculated by the forward propagation.



Gradient Descent

Gradient descent is an optimization algorithm used to train the neural networks.

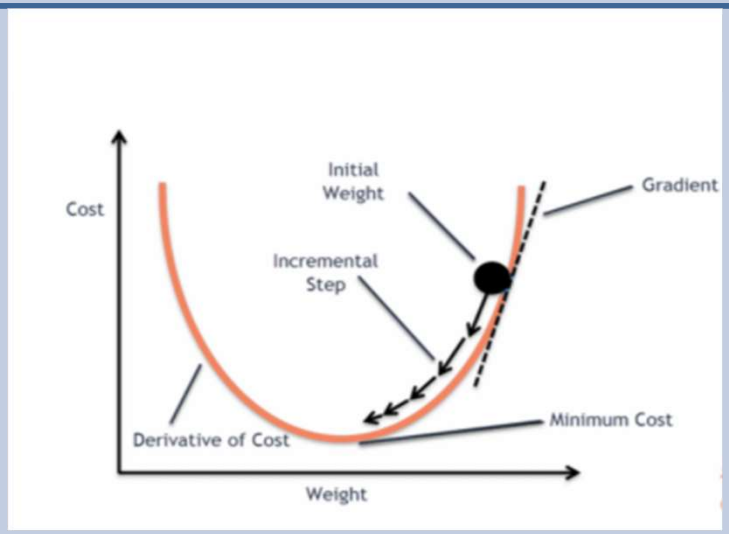
Goal is to minimize the cost function(sum of squared errors over the training set), by finding the optimal parameters of the model.

Optimal parameters will fit the data and give accurate predictions.

The cost function within gradient descent acts as a barometer, gauging its accuracy with each iteration of parameter updates.

Gradient Descent

Cost function graph



The graph shows the relationship between the cost function and the parameter(weight).

Gradient Descent

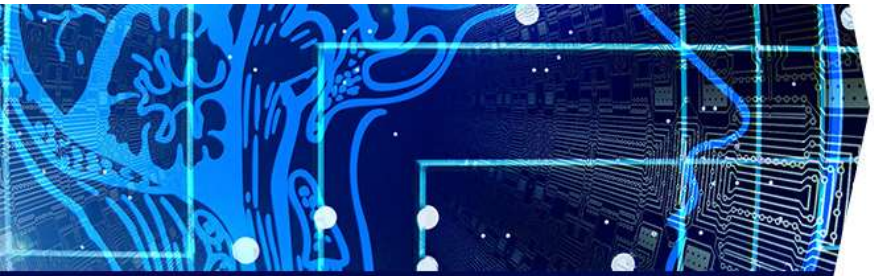
Cost function

The cost function measures the difference, or error, between actual y and predicted y at its current position.

This provides the feedback to the model so that it can adjust the parameters to minimize the error and find the local or global minimum.

It continuously iterates, moving along the direction of steepest descent (or the negative gradient) until the cost function is close to or at zero.

At this point, the model will stop learning.



Gradient Descent process

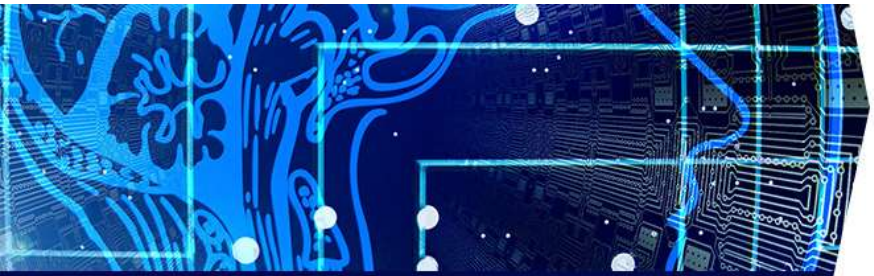
Initialize values for the coefficients for the function. These could be 0 or a small random value. So, we make coefficients = 0

The cost of the function is evaluated by plugging the coefficients into the function i.e., we do cost = f(coefficients)

The derivative of the cost function is calculated. We need to know the slope so that we know the direction (sign) to move the coefficient values in order to get a lower cost on the next iteration. So, we calculate, change = derivative(cost)

Now that we know the downhill direction from the derivative, we can now update the coefficient values. Specify a learning rate that controls how much the coefficients can change on each update. So, we do, coefficient = coefficient — (learning rate * change)

We repeat this process until the cost is 0 or close to zero.

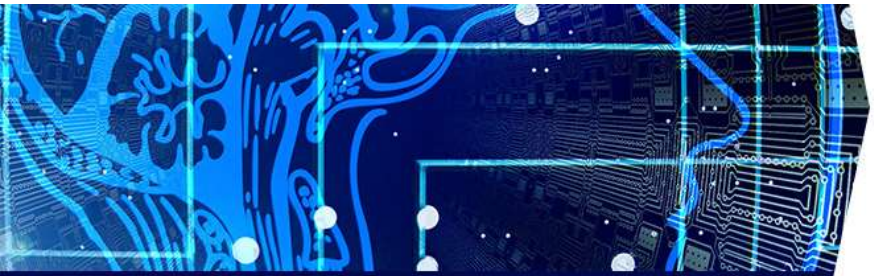


Deep Learning

Advantages

Following are the advantages of Deep Learning:

- Features are automatically deduced and optimally tuned for desired outcome. Features are not required to be extracted ahead of time. This avoids time consuming machine learning techniques.
- Robustness to natural variations in the data is automatically learned.
- The same neural network based approach can be applied to many different applications and data types.
- Massive parallel computations can be performed using GPUs and are scalable for large volumes of data. Moreover it delivers better performance results when amount of data are huge.
- The deep learning architecture is flexible to be adapted to new problems in the future.



Deep Learning

Disadvantages

Disadvantages of Deep Learning

- It requires very large amount of data in order to perform better than other techniques.
- It is extremely expensive to train due to complex data models. Moreover deep learning requires expensive GPUs and hundreds of machines. This increases cost to the users.
- There is no standard theory to guide you in selecting right deep learning tools as it requires knowledge of topology, training method and other parameters. Hence, it is difficult to be adopted by less skilled people.
- It is not easy to comprehend output based on mere learning and requires classifiers to do so. Convolutional neural network based algorithms perform such tasks.

Questions?

Thank you!